D4.1 Specification of Functional Architecture

Project number:	216541
Project acronym:	MULTI-BASE
Project title:	Scalable Multi-tasking Baseband for Mobile Communications
Start date of the project:	01/2008
Duration:	36M

Deliverable type:	Report
Deliverable reference number:	216541/ D4.1 / 1.0
Deliverable title:	Specification of Functional Architecture
WP contributing to the deliverable:	WP4
Due date:	M09
Actual submission date:	October 27th, 2008

Responsible organisation:	ULUND
Authors:	ULUND (Isael Diaz, Ove Edfors, Viktor Öwall, Farzad Foroughi) EAB (Thomas Olsson, Leif Wilhelmsson, Michal Stala) IFAT (Franz Dielacher, Peter Singerl) IMEC (Lieven Hollevoet, Liesbet v. d. Perre) LiU (Dake Liu) KUL (Wim Dehaene) TEC (Taru Kankkunen)
Abstract:	This document contains the initial definition of the Multi-Base functional architecture. The chosen modularity, description of the each block and the interconnection between the blocks are presented.
Keywords:	Functional architecture, key blocks

Dissemination level:	Public
Revision:	FINAL | 1.0

Instrument:	STREP
Thematic Priority:	ICT

Table of Contents

1. Executive Summary

This deliverable presents a functional architecture that presents the partitioning of the baseband part of the receiver into a set of functional blocks. A distinction can be made directly between control functionality and baseband processing, where the latter is the part requiring high processing capabilities. The baseband processing consist of several blocks with different demands and some can be executed on programmable blocks while other demand hardware accelerators.

We also do a brief description of each functional block and try to divide into ones we consider relatively mature and the ones we consider the challenges for the Multi-Base project. We introduce the concept of concurrent data streams which will increase the requirements on the processing capabilities, as well as the control functions, considerably.

The presented functional architecture will be used as the basis for the continued work within Multi-Base even though we consider this not to be the final architecture but a starting point for further exploration. Final selection of the programmable baseband processor has not been made. This function is considered a virtual module. The final decision is expected to be taken on the general assembly, November 19-20.

2. Introduction

This document serves as a general description of how the Multi-Base architecture looks like from a functional perspective. The architecture presented here is defined to provide all required functionality extracted from the use cases previously defined in D1.1 System Requirements Specification [1]. Even though a functional approach is discussed within this document, implementation aspects and constraints are taken into consideration when defining the various functional blocks and their correspondent boundaries.

In this document the system granularity is analyzed, by means of defining whether it is convenient to have a coarse grain architecture where a task is performed by a single unit through several internal operations or a fine grain approach where the same task is performed by many small components communicating intermediate results. The perfect balance between coarse and fine granularity would give the best suitable candidate in terms of flexibility, scalability and efficiency.

Flexibility is defined as the capability of being malleable, in other words, the capability to easily adapt to changes, which in this context implies that the architecture has to be able to adapt to dynamic changes. By designing a scalable platform, the architecture is defined so that extra extensions can be included in later stages in order to support future functionality, while being area and power efficient. The trade-off between these characteristics will make the architecture defined in this document a suitable platform for Multi-Base.

The three radio technologies that are considered have similarities and differences (See [1]). The functions that are common between the radio access technologies (RATs) have to be placed in re-usable blocks, while the functions that are different would be placed in configurable blocks or into specific purpose accelerators in order to provide the functionality required under the minimum area and energy cost. This document serves as a start in the investigation on these issues that will be considered throughout the project.

Bottlenecks have to be identified and proposals on how to address these will be presented. Accelerators may be an alternative for the functions with stringent timing limitations and general processing elements for those with more relaxed timing requirements.

The Multibase consortium members will focus on some specific functions that are identified as key research opportunities for the project. These are: Rx DFE, Tx DFE, Rx Demodulation, Rx Synchronization, Rx Channel estimation, and baseband processing. In this document, these elements are classified given their importance into research blocks. The other functions required for completing the transmission or receiving chains are out of the research scope of Multi-Base and are classified as available blocks.

The baseband processor is foreseen as a key element in the platform, since is functionality directly interacts with the rest of the platform. Decision about what kind of baseband processor would be used has not been taken at the time of producing this document. For the sake of clarity this functional model would be considered to be a virtual block throughout the entire document

Inter-connectivity between blocks is also addressed in this architecture, since it is relevant for proper functionality of the entire system. Inter-blocks synchronizations it is also considered as all the blocks or functions have to process data in a specific order.

The remaining parts of this document include Chapter 3, with a brief overview to OFDM and MIMO. Since both technologies play an important role in the definition of the architecture some basic concepts need to be introduced. Chapter 4 shows the architecture requirements extracted from the use cases and chapter 5 presents a view of a generalized functional model. Following in chapter 6 the functional architecture is presented and the blocks are described is some more detail. A future research approach is proposed in chapter 7. Some conclusions are presented in chapter 8. Finally list of abbreviations and references are listed in chapters 9 and 10.

3. A Brief Overview of OFDM and MIMO

In D1.1 the reason to choose only OFDM standards within our scope was motivated. Therefore, a brief introduction of what OFDM consists of has to be presented. This chapter will introduce some characteristics of OFDM in order to provide theoretical concepts that will be addressed in later chapters in a more technical manner.

OFDM divides the available spectrum into several subchannels or subcarriers. By making the subchannels narrowband they can be considered as frequency flat. In order to obtain a high spectral efficiency, the frequency response of the subchannels is overlapping but orthogonal. The inclusion of a cyclic prefix helps to maintain the orthogonality even in time-dispersive channels.

There are several reasons why OFDM schemes are very popular. Among those we can mention that OFDM is able to handle severe channel conditions like narrowband interference and frequency selective fading, without requiring a highly complex receiver

From the receiver's point of view, the use of a cyclic prefix longer than the channel's impulse response will transform the linear convolution in the channel to a cyclic convolution, described by the following equations:

$$y_l = FFT(IFFT(x_l) * g_l + \tilde{n}_l) = FFT(IFFT(x_l) * g_l) + n_l,$$

Where y_l contains the *N* received data points, x_l the transmitted constellation points, g_l the channel impulse response of the channel (padded with zeros to obtain a length of *N*), and $\tilde{n}_l$ the channel noise. Since the channel noise is assumed white and Gaussian, the noise after the receiver-side FFT is still white and Gaussian. We use the fact that the *FFT* of the two cyclically convoluted signals is equivalent to the product of their individual *FFTs.* Then the last expression can be expressed as:

$$y_l = x_l \cdot FFT(g_l) + n_l = x_l \cdot h_l + n_l,$$

Where $h_l = FFT(g_l)$ is the frequency response of the channel. Therefore the same type of parallel Gaussian channels as for continuous-time model is obtained (See [2]). Figure 1 and Figure 2 show a visual representation of the OFDM transmission over time and frequency.

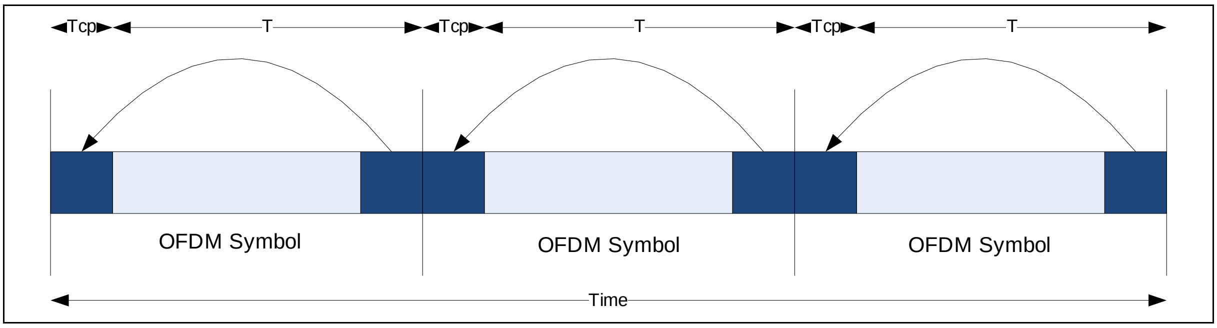

Figure 1: OFDM in Time

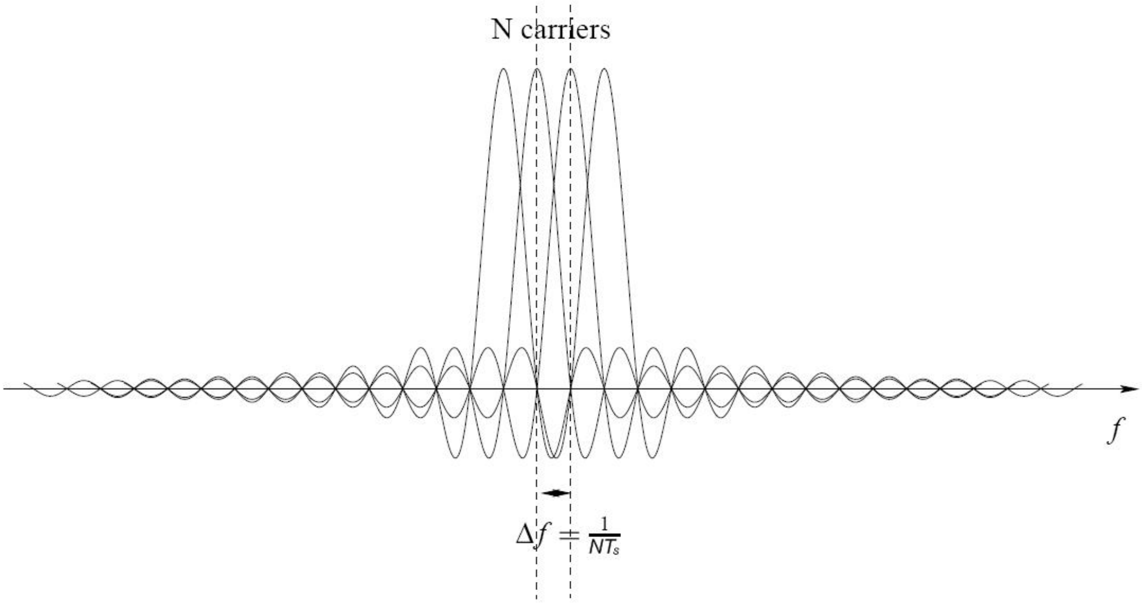

Figure 2: OFDM in frequency

The complexity of an OFDM system does not grow as fast as the complexity of a single carrier system with an equalizer as a function of the sampling rate [3]. This makes it easier to implement modems in hardware.

Nevertheless there are some drawbacks by using OFDM, among them it can be mentioned that OFDM does not capitalize on channel diversity without proper coding, which prohibits the use of uncoded OFDM schemes in fading environments. The diversity achieved by the OFDM system can be less than a single-carrier system employing the same error control code in a signaling environment rich in diversity. However, it has been shown that with proper coding, an OFDM system can exploit all the available diversity in the fading channel [3].

Even though OFDM schemes also suffer from disadvantages they provide by today's means one of the best alternatives for building high-speed spectrally efficient communications in the modern world. In a world where everything is becoming wireless and devices have to be powered by batteries, a technology that simplifies the receiver complexity and provides a high spectral efficiency is definitely a good alternative for future communications.

Another clear trend is that OFDM transmission technology, in various forms, are combined with multiple-input multiple-output (MIMO) technology to further enhance system performance. This latter addition does not readily support low-complexity implementations, but the promises of increased reliability through spatial diversity and/or increased data-rates through spatial multiplexing makes it a very interesting complement to OFDM. The two techniques work well together, in the sense that OFDM can assist in achieving reasonable complexity MIMO transmission/reception in time-dispersive channels. Both LTE and IEEE802.11n exploit this MIMO/OFDM combination to enhance the supported data rates.

4. Architecture Requirements

In the document D1.1 the general requirements were identified. The hardest use case to support is Use Case 4: Concurrent use of two standards for a long time, which therefore will impact the architecture in terms of required processing capability, memory etc.

Moreover, it was emphasized that the architecture should be able to handle the low bit-rate use cases with similar performance as low complex implementations specifically targeting only these use cases. This implies a highly scalable and reconfigurable architecture. Ideally, this will yield an architecture that can be used to address different market segments; from the low cost entry level segment to the most advanced “feature phone” segment.

In order to achieve this scalable architecture, it is considered essential to have a close interaction between algorithm development and architecture design. Moreover, algorithm/architecture co-design is deemed necessary to ensure that the time spent on implementation is used on the blocks where optimization really makes a difference.

Related to this is the ability to handle low-power modes efficiently. For instance scanning for new devices and scanning for available signals when being in idle mode.

Finally, to address that more frequency bands will be used in the future and that more communication systems will be interfering with one another, it is desired to have an architecture which allows for flexible spectrum allocation and implementation of detection of activity in the selected radio channel.

5. General Functional Model

In this section a general functional model will be outlined that will be used when assigning the considered standards to processing engines. Primarily we partition the system into control functionality and baseband processing. Control functionality is considered to have low requirements on processing capacity but is crucial to achieve a functioning system. Control functionality is required to run a single standard but is even more important when several standards are to be mapped onto a single platform, especially if concurrent operation is considered. Baseband processing involves several different blocks such as DFE, synchronization, modulation and coding. These blocks have very different complexity and therefore require different calculation capacity to achieve sufficient throughput. In this context we assume a programmable baseband processor suitable for low MIPS intensive algorithms while the real bottlenecks are to be mapped onto dedicated accelerators. Furthermore, we consider some of those computational modules to be reasonably well developed and solutions can be found within state-of-the art, e.g. FFT for OFDM processing, while for others considerable research is still needed, e.g. MIMO processing.

Memory requirements are also a crucial parameter in many systems of today because the memories are taking a considerable part of silicon area and power consumption. This statement is valid for all different parts of the architecture. In this document we will also briefly consider interfaces and communication requirements between different architectural blocks.

5.1. *Control*

Control processing has as previously stated low processing requirements. However, depending on the purpose the latency might be a crucial parameter. This will be exemplified by the following: we assume that the baseband processing units are in sleep mode to save power whenever they are not processing any data. In the case of for instance LTE, the packets are scheduled in time and the receiver knows when a packet is arriving. Hence the wake-up can effectively be scheduled by a programmable control unit, e.g. a micro processor. However, in the case of IEEE802.11n the arrival of data can take place at any time and the wake-up has to have very low latency and will require a hardware mapped control unit, e.g. an FSM. The control will also keep track of what sessions are active at any given time and in what mode they are operating.

Considering this we see that control functionality will be implemented as a combination of small hardwired control units for fast response and a programmable control unit which will provide flexibility.

5.2. *Baseband Processing*

The baseband processing consists of several blocks that will be described in chapter 6 and that have also been discussed in deliverable D1.1 System Requirements Specification. These blocks will be partitioned onto both programmable baseband units, which provide flexibility, and dedicated accelerators, which provide high throughput. This partitioning will be one result of the project since it highly depends on the level of concurrence envisioned and what data rates are expected in both single and concurrent mode. Executing a single standard might run on a programmable platform while a higher level of concurrency might call for dedicated solutions. The separation of algorithms between programmable and dedicated hardware solutions is one main consideration of Multi-Base.

5.2.1. Programmable

The programmable baseband platform is a task that is mainly dealt with in WP3. However, the requirements and the interaction between this and accelerators are the interface between WP3 and WP4. From a WP4 perspective requirements of the different baseband processing blocks will be analyzed and a partitioning will be made depending on the programmable platform alternatives.

5.2.2. Accelerators

The main task in WP4 is to develop dedicated accelerators for the concurrent signaling of the standards LTE, IEEE802.11n and DVB-H. The straightforward solution would of course be to implement three separate receiver chains. However, from a hardware perspective this is considered an inefficient alternative due to a large cost in silicon area. A main concern is to evaluate the processing algorithms for the different standards and to make clear to what extent they can share the same processing platforms. In D1.1 an initial specification on which blocks the project will focus its WP4 efforts is given and they are further described in the next section.

5.3. Memory

Memory requirement is one of the main considerations of many implementations today since the memories consume considerable amount of silicon area and power. In this context all types of storage is considered, i.e. memories, buffers and intermediate storage. The requirements will be evaluated from a both a single standard perspective and for the concurrent processing alternative.

Within each functional block there are memory requirements depending on algorithmic properties. These can be minimized through established procedures for memory handling. From a system perspective buffers and FIFOs between functional blocks need to be evaluated together with the computational engines to achieve an efficient dataflow with minimum storage requirements. However, when discussing concurrent processing of several data streams other issues arise. Depending on if different standards are to be executed on separate functional blocks or on a common flexible unit, different alternatives will need to be evaluated. In the case of separate blocks they can be treated with the regular methods but the total memory requirements will be correspondingly larger. If a flexible platform is to be used, the memory requirements will depend considerably on how the different streams will be scheduled in time, since it might be required to store intermediate results when switching between standards. Evaluating the consequences of the different solutions will be a key aspect within Multi-Base.

5.4. Communication

In this deliverable interfaces will not be specified. However, it will be considered what signaling is required and between which blocks it should be preformed. Latency requirements on signaling will also be considered.

6. Functional Architecture

In this section a generalized system overview is presented, describing which blocks are part of the baseband architecture. In section 6.2 a resume is given on blocks that are considered mature and that will not be the primary focus in Multi-Base. The following section presents the blocks which are the main focus of the Multi-Base project. It is envisioned that a large number of these blocks, if not all, will have to be implemented as hardware accelerators. However, this partitioning is a main concern of Multi-Base and hence a programmable baseband processing block is also being considered, as it is a natural part of any such system (see section 5).

6.1. System Overview

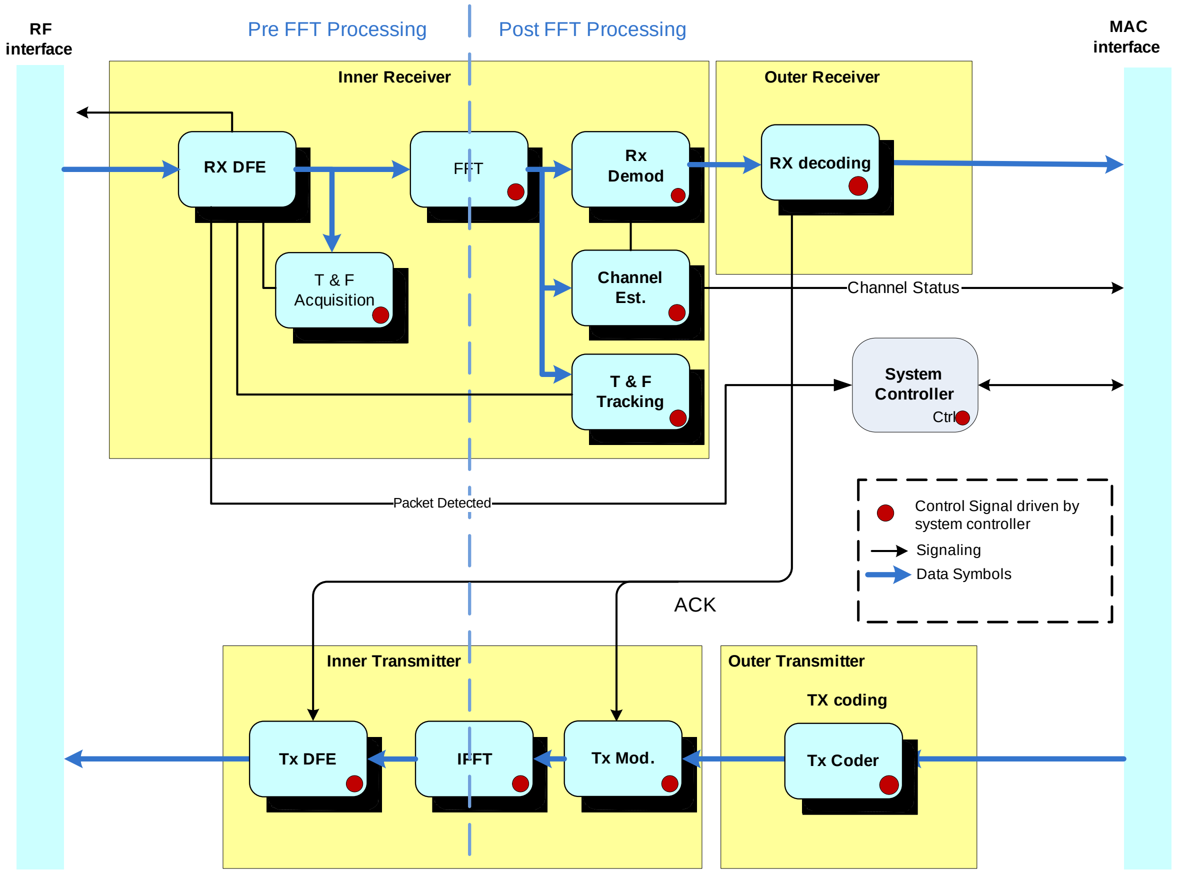

Figure 3: Functional View (All standards[1])

Common functional blocks are identified during the analysis of the selected standards (LTE, WLAN and DVB). The architecture is shown in Figure 3. For concurrent transmission and reception, each block is able to handle multiple data streams. At the moment of defining a common architecture, it is desirable to have a common functional architecture that is able to comply with the most challenging use case defined in D1.1 (use case 4 multi-standard concurrent support during long period of time). The architecture should be able to support several antennas, and consequently multiple data streams. However, if the investigation to be carried out during this project results in the conclusion that complexity or HW

---

[1] Transmission chain is neglected in DVB standard

requirements prohibit a practical implementation, various alternatives will be explored. An example could be that e.g. specific parts of the architecture would be implemented as a tile that supports any standard, but only one data stream, etc.

The following subchapters introduce a brief description of the contents of each block in order to present a holistic image of the entire system. More detailed information about the blocks is presented in 6.2 and 6.3.

6.1.1. Short Description of the Functional Blocks

Reception chain

The RX DFE contains A/D-conversion, radio control/feedback, downsampling/decimation, digital filtering, frequency compensation, and packet detection. It may also contain compensation for non-linearities detected in the channel estimator and removal of cyclic prefix. The DFE provides the means of inter-connection and communication between the RF circuitry and the baseband processing elements. Since it is placed first in the baseband processing chain its accuracy considerable affects the performance of the entire system.

In an OFDM system, after packet detection, decimation, filtering etc. it is required to switch the signal to the frequency domain which is achieved by a Fourier transform, most often the Fast Fourier Transform (FFT). The FFT marks the division between the time and frequency domain of the signal, or sometimes called pre-FFT and post-FFT stages.

In all communications systems synchronization is necessary, which can be performed in many different ways. In this case synchronization is divided into two stages coarse synchronization and fine synchronization, in other words: acquisition and tracking. These two stages have to be performed for synchronization in time and frequency respectively. The time synchronization strives to find the best starting points for the FFT while the frequency synchronization estimates the frequency error to compensate for remaining frequency mismatch after the AFC.

The channel estimator uses the received pilot symbols to provide channel estimates to the channel equalizer. The block can also provide other estimates, such as time and frequency error estimates to the synchronizer, CQI estimation, Doppler and delay spread estimation.

The Rx demodulation block contains a channel equalizer and a demapper to map symbols into data bits. This block takes the data from the FFT and the channel estimate from the channel estimator and tries to correct the errors according to the channel conditions detected. These operations try to correct the errors in the information assuming that the channel does not change dramatically during an OFDM symbol.

Even though the data was equalized according to the channel estimates, it is likely that it still contains errors and that it might be impossible to recover the original transmitted data. For this reason coding is used to enhance the chances to successfully recover the data. The decoding block contains decoders such as Turbo decoder, Viterbi decoder and Reed-Solomon Decoder. It also contains support for retransmission, such as HARQ. More detailed information about which decoding scheme is used is presented in section 6.2.2.

The higher hierarchical element depictured in Figure 3 is the MAC layer, which in this case marks the end of the physical layer and consequently the end of this project’s scope.

Transmission Chain

Elements in the transmission chain follow the inverse order than in the reception chain; however the transmission chain is somehow simplified given that synchronization and channel estimation are not required anywhere else but in the reception.

Information from data link layer (OSI Model) is received from the MAC interface and introduced into the system depicted in Figure 3; a coding scheme is added to the information in order to provide better reliability on the information by the block named Tx coding.

Tx coding contains a coding scheme depending on which standard is being operated, since the three standards require different coding, i.e. Turbo encoder, convolutional encoder, block encoder and support for retransmission (HARQ).

The transmission modulator block contains mapping of coded bits to symbols and addition of a cyclic prefix. The operation of transforming the bits into OFDM symbols is done by applying a IFFT, however LTE uses a slightly different scheme than OFDM, namely SC-FDMA which as a consequence uses an FFT instead (A brief introduction to the standards was presented in D1.1).

The final operation before sending the OFDM symbols to the RF interface is performed by the DFE where operations like digital pre-distortion, interpolation; noise-shaping, cross-point estimation can be performed.

System Controller

The system controller controls the interconnection between blocks in the architecture, shuts downs blocks to save power, or wakes them up when they are required and keeps system timings and system status continuously. This module can contain contains both programmable blocks and hardwired FSMs as discussed in section 5.

6.2. Available Blocks

6.2.1. FFT

Scalable efficient FFT architectures have been researched for a long time. Within the Multi-Base project this block is available within the consortium and is foreseen as a mature research topic.

A flexible FFT architecture has been considered for OFDM in [4] and in [5] an FFT implementation, where a 2048-complex point pipeline FFT processor is designed based on Radix-2 and Hybrid floating point is presented. These are two examples where Multi-Base can take advantage of existing implementations that already contain the required flexibility needed to support a multi-standard concurrent implementation. However, updating of the architectures to fit the Multi-Base concept is most likely required.

Research in this area will be carried out only if it is proven that an FFT would need modifications in taking more than one standard into consideration.

6.2.2. Forward Error Correction

Forward Error Correction (FEC) is foreseen as one of the bottlenecks in the process of implementing each of the standards. Therefore, it is presumed that a specific accelerator would be necessary in order to code/decode each of them. Under this assumption it is clear that three different accelerators are contained underneath our FEC block definition, where each accelerator can operate in parallel in relation with the others. Table 1, extracted from D1.1[1], shows the different FEC accelerators required per RAT.

Standard	**802.11n**	**3GPP – LTE**	**DVB-H**
Coding	Convolutional, g=133, 171 R=1/2, 2/3, 3/4, 5/6 LDPC Block length 648, 1296, 1944	Convolutional or duo-binary turbo	Inner: Punctured convolutional g=133, 171, CR =1/2, 2/3, 3/4, 5/6, 7/8, Outer: Reed-Solomon (204, 188)

Table 1: Standard's Coding Scheme

Efficient FEC implementations for the various RATs are already available, and therefore research within this field it is placed outside Multi-Base interests, for specific examples of existing implementations consult [6], [7], [8], [9], [10] and [11].

6.3. Research Blocks

6.3.1. DFE Rx

Overview

A DFE in an OFDM system is a functional block between the analog RF (analog frontend), and the RX demodulator on the other side. It contains functional blocks that implement frequency compensation, decimation, pre-filtering, digital downsampling, AGC and coarse time synchronization. It also contains the frontend data and control interfaces. The design goal of the DFE RX is to support processing samples at 100 Msamples/s at the analog frontend interface, while consuming little power when no packets are being received (sensing-mode).

Taking the functions performed in the DFE as modules where data and control are split into two different paths, data-path and control path can be distinguished in Figure 4.

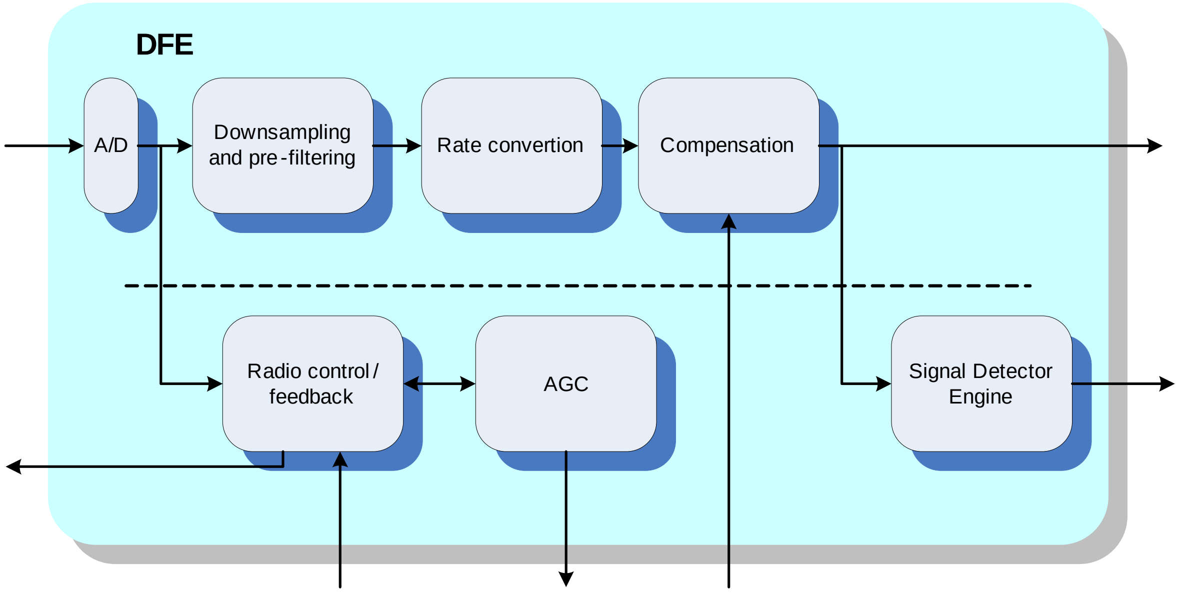

Figure 4: DFE Functional View

Hardware Description

The DFE allows monitoring of signals coming from the antennas and allows performing the wakeup sequence of the platform in case a valid packet is being received. An example of a DFE (See Figure 5) consists of a controller, a detection engine, downsampling filters and a buffer. In this exemplary illustration, specific numbers are given concerning memory requirements and number of bits required. These are provided to give a feeling for the size of the DFE, rather than putting limits to the developments that will be done within the project.

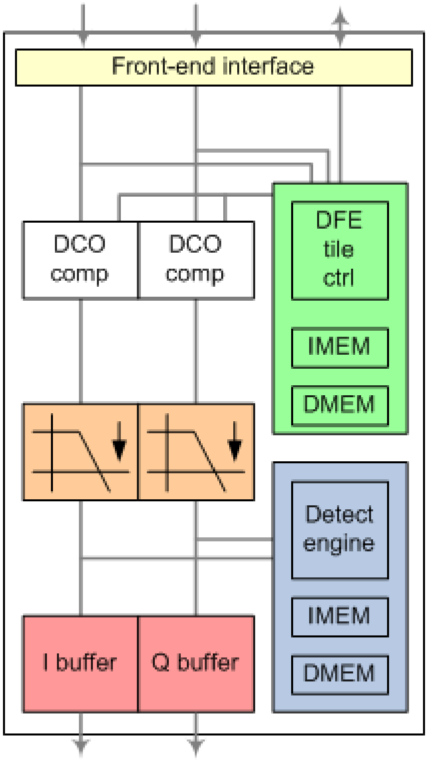

Figure 5: DFE Architecture

The low-power microcontroller features a reduced instruction set and takes care of signal power detection, frontend control (AGC) and the power management of the other blocks. The power management is implemented as a hierarchical wakeup: upon detection of power on the incoming signal, the controller sets the frontend gain and enables the downsampling filters, buffers and coarse time synchronization. When sync is found, the synchronization engine is disabled and an interrupt is generated towards the rest of the platform. The digital downsampling filters are power-optimized decimation filters.

Coarse time synchronization is mapped onto an ASIP that is optimized for signal detection purposes. The ASIP has 5-issue VLIW architecture. Two scalar slots each contain a 16b data path for address computations and control. The 3 vector slots are 128b vector units, capable of processing 4 input samples every clock cycle. The ASIP also contains a vector scratchpad and an instruction memory. This ASIP can also be used to implement sensing of the selected radio channel.

Memory and communication constraints from the example in Figure 5 are presented in the following subsections. Based on these requirements, estimations can be made by taking into consideration that multi-standard execution is addressed.

Memory Requirements

- Microcontroller: instruction memory 14 Kbit, data memory 0.5 Kbit

- ASIP: instruction memory 64kbit, data memory 64kbit
- Buffers: 48 Kbit, or larger depending on the architecture of the rest of the system

Communication Requirements

- Frontend interface for data: 2 x 12 bit inputs for the I and Q input samples @ 100 Msamples/s
- Frontend control interface: 12 bit address outputs, 16 bits bidirectional data bus, enable signal, read/write signal, ready input

- Data output: bus supporting 24 or 32 bits output @ 50 MHz
- DFE RX control interface: bus system to support programming the controller and the ASIC program memories and general parameters (to be refined)
- Interrupt output

Challenges

Due to its functionality and complexity, the DFE is a niche of opportunities for research development, some examples are presented as follows.

Radio control/feedback
Power control of the radio depending on the signal quality in the system could be investigated more, finer granularity, power management across various error conditions, e.g. when no error is present the power can be lowered. A similar approach could be considered in clocking, e.g. when having good signal conditions fewer bits could be used as input to the system. Lowering the frequency on a sigma-delta A/D converter gives fewer bits and potentially lowers the power consumption.

An AGC is used to achieve the best possible signal to noise ratio at the ADCs. Functionalities like AGC and AFC are typically seen in a DFE. Research work has been done in the AGC area [12], [13] (WLAN). Further research could be done in the multi-standard aspect.

The UE must be able to switch between standards. The radio must be controlled to be able to select the different frequency bands. There might also be need for sensing/scanning for available signals. The radio control mechanism should also be a part of the DFE.

Digital power control
A power control unit could be a part of the DFE in burst based wireless communication technologies, like WLAN [14]. The DFE contains a block for signal detection and controls power for the subsequent blocks.

Downsampling by decimal number – simple pre-filters (CIC, WDF)
A typical functionality that is part of a DFE, especially in multi-standard solutions, is Downsampling/decimation and rate conversion (See [15], [16] and [17]). The authors describe a DFE supporting a system with three standards: GSM, IS-95 and UMTS. Note that the Multi-Base project focus on OFDM systems (LTE, WLAN and DVB-H) and none of the standards mentioned above is OFDM based. The first steps in [15], [16] and [17] are the decimation steps. This is done to down sample the input signal to roughly the desired sampling frequency (normally 2 or 4 times the symbol rate). The use of CIC and WDF are used to accomplish the decimation. Advantages with these filters are that they are very simple – there are no multipliers used. The disadvantage is that the filters are not optimized. In order to compensate for the poor performance, drooping band-pass and wide transition

regions, a FIR filter is often applied to compensate for the losses [18]. Investigation on improvements/alternatives of these filters could be a part of the research in this project. The OFDM context should be considered.

Rate conversion (FSRC)
To be able to get the desired frequency, after the decimation, a FSRC could be used as proposed in [15], [16] and [17]. Investigation on a design of a general and flexible FSRC could be a part of this research project. An investigation of a combined FSRC unit that both corrects for sampling error but also re-samples to the desired frequency could be another research area in this project. Alternatives to the Farrow FSRC should also be considered.

Compensation
Frequency compensation could be done in the analog or digital domain. If it is done in the digital domain it could be considered as an interaction between the time-frequency tracking and the DFE block. In some cases, especially in OFDM systems, estimation in frequency domain and compensation in time domain could be preferred. Timing Compensation is performed in DFE as well, taking the estimation from the synchronization blocks as seen in Figure 4.

Another compensation that can be treated in the DFE is IQ imbalance; in fact IQ imbalance estimation and correction could be done both in the frequency and in the time domain. One system could be designed where estimation is done in the frequency domain and compensation in time. Another solution would be to have both estimation and correction in the frequency domain. Any combination (4 in total) of these blocks and domains could be possible, However, if this compensation is to be performed in the DFE, it will have to be performed in the time domain given the position of the DFE in the reception chain. IQ imbalance arises from mismatches in the I and Q paths in the radio front end. The IQ imbalance is usually modeled as an error in amplitude and an error in phase for the IQ paths. The removal of the IQ imbalance is usually done in two steps: IQ imbalance estimation and IQ imbalance correction. There are various ways of arranging the estimation and the correction part in an OFDM system.

6.3.2. Rx Demodulation

The RX demodulation block takes the received channel values and perform a demapping, using the channel estimates, thus compensating for channel phase and amplitude. Since the channel will vary in strength both across subcarriers and in time, this block will also deliver reliability for each symbol so that the subsequent decoding has sufficient information to perform well. The details on the representation of the output from this block, word lengths etc. will be determined through analysis and simulation, where different options are compared.

6.3.3. Rx Synchronization

The downlink synchronization block is considered to include both time and frequency synchronization. In OFDM systems it is crucial to know the exact start and end of the OFDM symbol. Time synchronization provides the OFDM system with correct timing of the OFDM symbols.

The oscillators in the base stations and in the user terminals are usually not ideal. Therefore, frequency synchronization between these two must be handled. Time and frequency synchronization compensation is usually done in the time domain. Frequency compensation could also be done in the analog domain. The estimation can be done in time or frequency domain or a combination of both.

Time and frequency synchronization in separate, individual OFDM systems has been a research topic for some years now. Investigation could be more targeted on how time and frequency synchronization is handled in a multi-standard solution. The synchronization symbols and pilot pattern are different in the three standards that the Multi-Base project is addressing. How can this be handled with a general solution?

Hardware Description

The synchronization block is part of the downlink chain and it is placed between the channel estimation and the DFE. The block is expected to take the estimation of the channel for the three RATs and interact with the DFE to maintain a good link with the corresponding base stations.

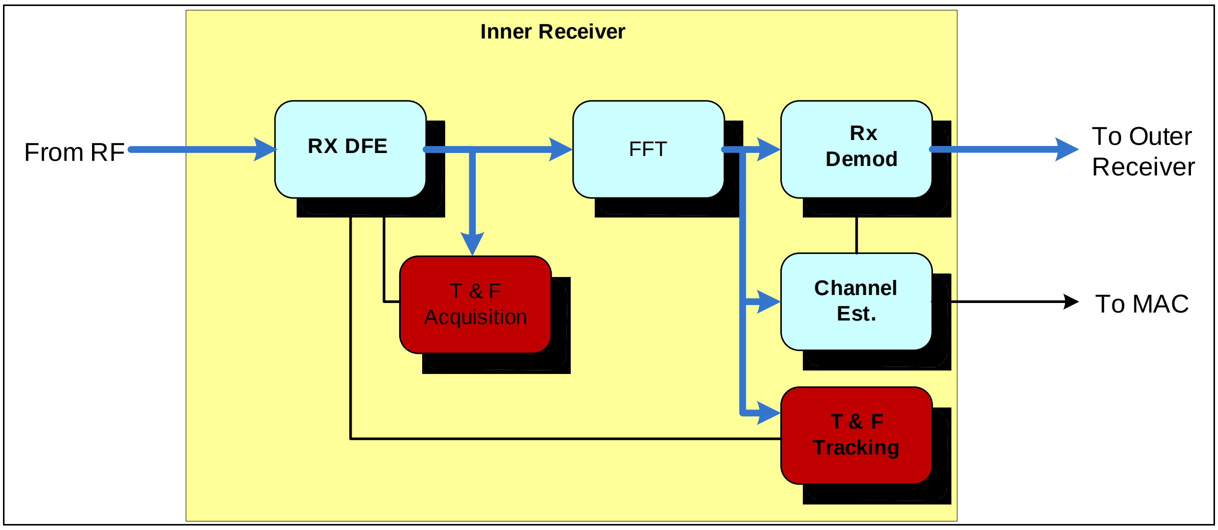

Figure 6: Synchronizer interaction in DL chain

Given that all RATs under analysis are OFDM, the cyclic prefix could be used for synchronization as proposed in [19], where the author estimates time and frequency offsets by correlating using the cyclic prefix. The principle is that if the FFT is placed in the part of the cyclic prefix that is not affected by the previous symbol due to channel dispersion (See Figure 7), the symbols obtained from the FFT will be affected by a phase shift proportional to the time offset, while the difference in phase shifts between subcarriers remains constant.

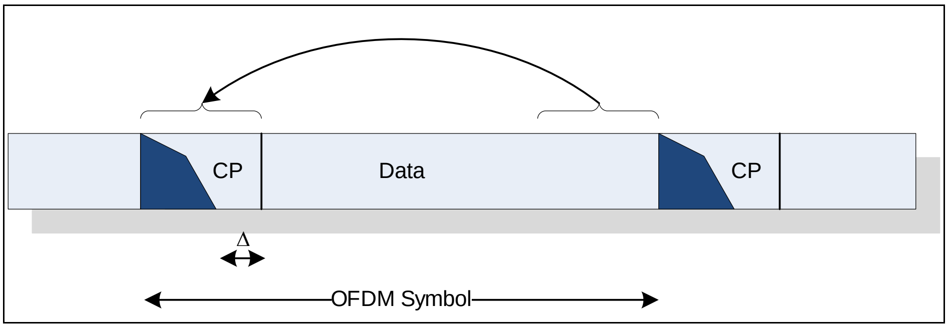

Figure 7: Cyclic prefix for synchronization

This technique can be implemented considering roughly the elements depictured in Figure 8.

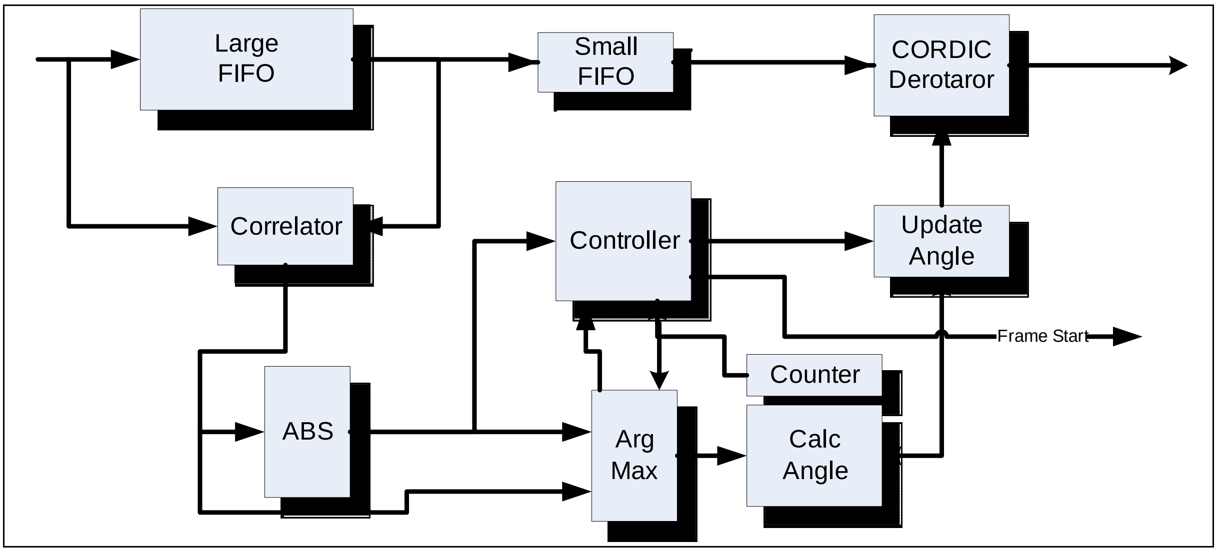

Figure 8: Cyclic Prefix Synchronizer Architecture

Memory Requirements

In [19] a general OFDM synchronizer for a single standard is presented. This solution consists of specific purpose design and requires around 32Kb of memory to perform synchronization at 130 MIPS, however, since our goal is to support three standards with maximum flexibility, memory requirements are expected to increase by a factor of larger than 3. This is a crucial fact since memory is the main contributor to the silicon area of the presented design. This implementation is however quite old and advances in technology should be investigated to increase the effectiveness.

Communication Requirements

The synchronization block expects a wake up signal from the Rx-DFE when a new frame on 802.11n is detected and a wake up signal from the control entity when a new frame or burst is detected from LTE and DBV respectively.

The synchronization block expects the channel estimation to provide a rough estimation of the channel in order to adjust the incoming data.

The synchronizer has close interaction with the DFE so synchronization is enabled only when needed, to detect whether there is more than one RAT sending information to the terminal, etc.

6.3.4. Rx Channel Estimation

The primary task of the channel estimator is to use the received pilot symbols to provide the channel estimates required for data detection. Depending on the standard and the specific signal constellation addressed, there will be different requirements on the quality of the channel estimates. Simply put, large signal constellations require channel estimates with low error variance, while small signal constellations can tolerate a larger error variance.

In situations where the signal constellation reflects the current channel condition, i.e. when adaptive modulation is used, large signal constellations will be used when the SNR is high. It is also in these high SNR cases that we can expect the variance on the channel estimate to be low. The quality of the channel estimate thereby has a tendency to automatically follow the changes on the requirements. This is, however, only true in the cases where we have a nice and proportional relation between input and output SNR of the channel estimator. Such a proportional relation does not always exist and some of the most promising estimator structures from a complexity point of view have an error floor behavior which limits their performance at high SNRs. The error floor is usually decreasing with increasing estimator complexity. To obtain a good trade-off between computational complexity and receiver error performance, we need to make in-depth analysis of all three standards and device a general strategy where we spend as little resources as possible on channel estimation in each situation, without sacrificing error performance.

In addition to its primary task, this block may also provide other estimates which fit well into its computing structure. This is an issue that will have to be investigated further, since it depends to a large extent on the chosen structure.

Challenges

DVB-T/H, LTE and IEEE 802.11n have their own specific pilot and preamble pattern. A consistent channel estimation based on numerous existing methods (e.g. Least Square, Minimum Mean Square Error, etc.) can be done to acquire the best Channel Impulse Response (CIR) for each of these standards. However, due to processing power limitations on the mobile terminal, CIRs might possibly be preprocessed for a number of scenarios and stored on memory. Later the appropriate CIRs are chosen from the pool of stored CIRs based on the Doppler frequency, etc. and used to track varying channel behavior ([20], [21]).

Thus, research will focus on finding the best channel estimates for these different standards based on the requirements of each. Besides, investigation will also focus on finding similarities in CIRs among these standards to increase optimality and reduce hardware cost as much as possible. It is also of prime interest to find an algorithm which can keep track of changes in channel behavior and update CIRs at an acceptable rate.

Meanwhile, it is desired to investigate the effect of limited precision mathematics on channel estimations. Varying signal precision representation as well as stored coefficients can provide interesting results and help us decide an optimum hardware/software to handle the arising issues.

6.3.5. DFE Tx

Hardware Description

Modern modulation formats for communication systems are usually designed for high spectral efficiencies. These lead in general to strong fluctuating signal envelopes with high peak-to–average power ratios (PAR), which usually limits the efficiency in conventional transmitter architectures. Radio frequency switched mode power amplifiers (SMPA) are potential candidates to increase the efficiency considerably because on an ideal switch the currents and voltages never occurs simultaneously. Because the SMPA can handle only a limited number of different states as e.g. “On” and “Off”, a digital modulator is required to transform a complex baseband signal (I and Q or magnitude and phase) with a certain amplitude resolution into a signal with only a few different magnitude levels (e.g. pulses with

"0" and "1"). To preserve the original information of the complex baseband signal some of the information will be encoded into the timing of the two level sequence (Also more than two levels are possible). For this reason a pulsewidth- or a delta-sigma encoder is needed within the digital modulation block. The hardware block that will be designed has as target to digitize the Tx front end as much as possible. The advantage of this strategy is that the flexibility towards multi-standard support is maximally increased.

The design of the power amplifier and the digital modulation block cannot be decoupled. The feasibility of the whole setup depends on the possible interaction between digital modulation and the SMPA. Furthermore the frequencies needed in the digital will, at least for the part close the PA, go into the GHz range. This will require circuit level digital design.

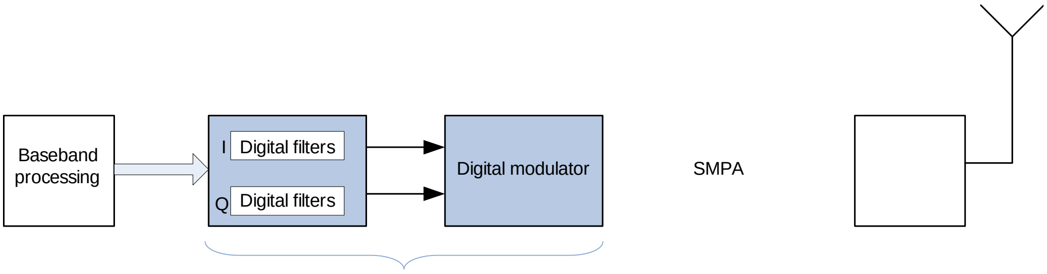

Figure 9:Block diagram Digital Transmitter

Figure 9 depicts a simplified block diagram of a switched mode transmitter (without feedback for pre-distortion) including the digital frontend. The first block after the baseband signal processing is composed mainly of digital filters to fulfil the spectral requirements on the transmission signal which is imposed by the regulatory bodies and some digital correction units (digital pre-distortion core, cordic,..) to compensate the non-idealities mainly caused by the analog RF circuits. This block should support multiple communication standards. The digital modulator in Figure 9encodes the incoming complex baseband signal into a digital sequence with two or more discrete amplitude levels to drive the SMPA (s) in an efficient way. To recover the bandlimited RF signal before we feed them to the antenna we need a bandpass filter to suppress the unwanted harmonics generated in the digital modulator.

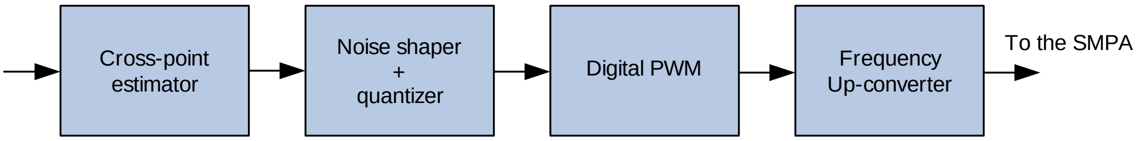

Figure 10: Block Diagram Digital Modulator (I or Q path)

Figure 10 depicts the I- or Q-path of the digital modulator from Figure 9, which is composed mainly of a digital pulsewidth modulator and a noise-shaper who shapes the noise introduced by the digital PWM out of the interesting frequency band. The cross-point estimator is needed to avoid a nonlinear distortion in the wanted frequency band caused by the digital uniform pulsewidth modulator. After encoding the baseband signal into a pulse sequence, the signal will be shifted up to the wanted carrier frequency.

Memory Requirements

This still needs to be studied but it will mainly depend on the amount of reconfigurability that is needed to support the different standards.

Communication Requirements

The communication requirements are modest

- On the baseband side a normal streaming interface is required. To reduce timing problems with other blocks a FIFO based interface is proposed
- The interface between the digital modulation and the switching PA is dedicated and will be designed in the project.

6.3.6. Programmable Baseband Processor and Control

Hardware Description

The programmable baseband processor shall manage miscellaneous and unstable baseband functions that cannot be supported by accelerators. The programmable baseband processor shall also manage high level control, module configurations, and inter-module connections. According to the System requirement specification from WP1 D1.1 project, latencies induced by control and baseband signal processing are the number one critical issue.

The proposal is to use a small general purpose DSP controller from LiU for control of system level connections and data dependencies on inter-accelerator level. The name of the DSP controller is “senior”, which does not support programmable baseband functions. Senior processor is under development by LiU.

There will be sufficient accelerator ports running in parallel. Moreover, multiple accelerators can be connected to one port if these accelerators do not need port access at the same time. (For example, different digital frond-end modules for different standards can be connected to one port.) The “senior” will offer control “configurations of accelerators” for all accelerators. The “connections for data communications” will be described in the section “Communication requirement” in this chapter.

Memory Requirements

According to the System requirement specification from WP1 of Multi-Base project (D1.1), data buffers in our baseband signal processing are small. There is therefore not necessary to have a main memory on top level. Each accelerator holds its own data memories. Each data memory in any accelerator shall be shared as a streaming (FIFO) buffer by other accelerators. Inter-accelerator function designer will decide the model of sharing (sharing address space and memory or sharing data transaction time). Up to 16 data connections (or 8 connection channels) in parallel shall be available for data sharing.

Communication Requirements

Connections for data communications can be implemented using any OCN (On-chip Connection Network). Senior DSP controller can send arbitration and connection control via one accelerator port.

7. Hardware Emulation

Alongside with a programmable baseband processor there is also the possibility to use hardware emulation techniques for architecture exploration and software debugging. For instance a transaction level approach based on SystemC can be used to develop a virtual platform. The virtual platform can model the system at the desired level, as an example with or without timing information.

One obvious drawback with an approach relying on hardware emulation is that the possibilities for demonstration are limited due to the lack of real-time performance, however there are several advantages obtained from using a virtual platform compared to FPGA or ASIC solutions, such as:

- Increased observability. Any variable can be observed at any time.
- A natural top-down approach with gradually refined level of abstraction for all or only a selected set of components.
- Decreased turn-around time. Many alternative solutions may be investigated during short time.
- Co-development of hardware and software. Bottlenecks found during software development can easily be addressed to obtain an optimized hardware solution.

8. Conclusions

In this deliverable a functional architecture has been presented and functional modules have been divided into what is considered relatively mature and to which we will focus our research efforts. These blocks have been considered in more detail which will be the starting point for the continued research within Multi-Base. The main focus will be on the Digital Front End (DFE), channel estimation and synchronization while other blocks will be investigated on requirement. For instance there is a long tradition of implementing FFT modules and these should be available with only minor modifications.

9. List of Abbreviations

3GPP	Third Generation Partnership Project
A/D	Analog to Digital
AFC	Automatic Frequency Control
AGC	Automatic Gain Control
ASIC	Application Specific Integrated Circuit
ASIP	Application Specific Instruction set processor
BER	Bit Error Rate
BPSK	Binary Phase-Shift Keying
BS	Base Station
CCK	Complementary Code Keying
CDMA	Code Division Multiple Access
CIC	Cascaded Integrator-Comb
CIR	Channel Impulse Response
CORDIC	COordinate Rotation DIgital Computer
CP	Cyclic Prefix
CQI	Channel Quality Indicator
DFE	Digital Front End
DL	Downlink
DMEM	Data MEMory

DSP	Digital Signal Processor
DSSS	Direct Sequence Spread Spectrum
DVB-T/H	Digital Video Broadcasting -Terrestrial/Handheld
DwPTS	Downlink Pilot Timeslot
EDGE	Enhanced Data rates for GSM Evolution
FDD	Frequency Division Duplexing
FEC	Forward Error Correction
FFT	Fast Fourier Transform
FIFO	First Input First Output
FLO	Forward Link Only (Mobile TV)
FPGA	Field-Programmable Gate Array
FSRC	Fractional Sample Rate Conversion
GERAN	GSM EDGE Radio Access Network
GNSS	Global Navigation Satellite Service.
GP	Guard Period
GSM	Global System for Mobile communications
HARQ	Hybrid Automatic Repeat-reQuest
HSPA	High Speed Packet Access
IFFT	Inverse Fast Fourier Transform
IMEM	Instruction MEMory

ISDB-T	Integrated Services Digital Broadcasting - Terrestrial
LS	Least Square
LTE	Long Term Evolution
MAC	Media Access Control
MBMS	Multimedia Broadcast Multicast Service
MIMO	Multiple Input Multiple Output
MIPS	Million Instructions Per Second,
MLME	MAC Sub-layer Management Entity
MMSE	Minimum Mean Square Error
NFC	Near Field Communication
OCN	On Chip Network
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open Systems Interconnection
PAR	Peak-to-Average Ratio
PAR	Power Amplifier
PHY	Physical layer of the OSI model
PLCP	Physical Layer Convergence Procedure
PLME	Physical Layer Management Entity
PMD	Physical Medium Dependant
PWM	Pulse Width Modulator

QAM	Quadrate Amplitude Modulation
RAT	Radio Access Technology
RF	Radio Frequency
SC-FDMA	Single Carrier Frequency Division Multiple Access
SDR	Software Defined Radio
SIMD	Single Instruction Multiple Data
SISO	Single Input Single Output
SMPA	Switched Mode Power Amplifiers
SNR	Signal to Noise Ratio
STBC	Space Time Block Coding
TDD	Time Division Duplexing
TDMA	Time Division Multiple Access
UE	User Equipment (user terminal)
UL	Uplink
UMB	Ultra Mobile Broadband
UML	Unified Modeling Language
UMTS	Universal Mobile Telecommunications System
UpPTS	Uplink Pilot Timeslot
UTRAN	UMTS Terrestrial Radio Access Network
UWB	Ultra Wide Band

WCDMA	Wideband CDMA
WDF	Wave Digital Filter
WiMAX	Worldwide Interoperability Microwave Access
WLAN	Wireless Local Area Network
VLIW	Very Long Instruction Word
VoIP	Voice over IP
WP	Work Package

10. References

[1] MULTI-BASE Consortium, Deliverable D1.1 First version, System Requirements Specification, August 2008

[2] Ove Edfors, “Low-complexity algorithms in digital receivers”, Doctoral Thesis, Luleå University of Technology, Sept 1996

[3] M Debbah, “Short Introduction to OFDM”, White Paper, Mobile Communications Group Institute Eurecom, 2004

[4] F. Kristensen, P. Nilsson, A. Olsson, ”Generic Transmitter for Wireless OFDM Systems” 14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications (PIMRC), 2003.

[5] T. Lenart, V Öwall, “Architectures for Dynamic Data Scaling in 2/4/8K Pipeline FFT Cores”, IEEE Transactions on VLSI,Nov 2006.

[6] Rovini, M.; L'Insalata, N.E. ;Rossi, F. ;Fanucci, L., "VLSI design of a high-throughput multi-rate decoder for structured LDPC codes", 8th Euromicro Conference on Digital System Design, Sept. 2005

[7] C Tseng, CC Lin, HC Chang, CY Lee, "A power and area efficient multi-mode FEC processor", ISCAS'04

[8] Yao, J.; Wei-Feng Huang; Ming-Syan Chen; " DFEC: Dynamic Forward Error Control for DVB-H", International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006.

[9] Javier Gonzalez-Bayon, Carlos Carreras, Angel Fernandez-Herrero, “A Comparison of Frequency Offset Synchronization Algorithms for WiMAX OFDM Systems”, EUROCON 2007.

[10] Bougard, B.; Giulietti, A.; Van der Perre, L.; Catthoor, F.; “A class of power efficient VLSI architectures for high speed turbo-decoding”, Conference on Global Telecommunications, 2002

[11] M. Kamuf, V. Öwall, J. B. Anderson, ”Optimization and Implementation of a Viterbi Decoder under Flexibility Constraints”, IEEE Transactions on Circuits and Systems - I: Regular Papers,Vol. 55,No. 8, pp. 2411-2422, 2008.

[12] Emilija Mileti, Miloš Krsti, Maxim Piz, and Michael Methfessel, “Digital Automatic Gain Control Integrated on WLAN Platform”, PROCEEDINGS OF WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY VOLUME 31 JULY 2008

[13] Peter Singerl, Christian Vogel, “A Fast and Accurate Automatic Gain Control For a Wireless Local Area”

[14] Bruno Bougard, Lieven Hollevoet, Frederik Naessens, Thomas Schuster, Ching Ho, Liesbet Van der Perre, “A low power signal detection and pre-synchronization engine for energy-aware software defined radio”, Proceeding of the SDR 06 Technical Conference and Product Exposition

[15] G. Hueber, L. Maurer, G. Strasser, K. Chabrak, R. Stuhlberger, and R. Hagelauer, ”On the design of a multi-mode receive digital-front-end for cellular terminal RFICs”, IEEE 2006.

[16] G. Hueber, L. Maurer, G. Strasser, K. Chabrak, R. Stuhlberger, and R. Hagelauer, ”A digital-front-end for multi-mode/multi-system capable receivers for cellular terminals” IEEE, 2006.

[17] G. Hueber, L. Maurer, G. Strasser, K. Chabrak, R. Stuhlberger, and R. Hagelauer, ”A digital-front-end for multi-mode/multi-system capable receivers for cellular terminals” IEEE, 2006.

[18] Richard Lyons, “Understanding cascaded integrator-comb filters”, Article published in electronic magazine embedded.com published 03/31/05, 02:49:00, available online in http://www.embedded.com/

[19] S. Johansson, D. Landström, P. Nilsson, ”Hardware Architecture of a General OFDM Synchronizer”, Proceedings of the 10th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 99), Osaka, Japan, 1999.

[20] Jan-Jaap van de Beek, Ove Edfors, Magnus Sandell, “On Channel Estimation in OFDM Systems”, 45th Vehicular Technology Conference, July 1995.

[21] Bougard, B.; Giulietti, A.; Van der Perre, L.; Catthoor, F.; “A class of power efficient VLSI architectures for high speed turbo-decoding”, Conference on Global Telecommunications, 2002